

IN THE CLAIMS:

The text of all pending claims, (including withdrawn claims) is set forth below. Cancelled and not entered claims are indicated with claim number and status only. The claims as listed below show added text with underlining and deleted text with ~~striethrough~~. The status of each claim is indicated with one of (original), (currently amended), (cancelled), (withdrawn), (new), (previously presented), or (not entered).

Please AMEND claims 1, 5, and 9-10 in accordance with the following:

1. (Currently Amended) A method of executing markup document applet by a browser, comprising:
receiving a request for executing an applet from the browser;
determining whether the applet is a bound applet or an unbound applet;
loading the requested applet into a virtual machine; and
if the applet is ~~an~~ the unbound applet, immediately issuing predetermined commands to the virtual machine to first set the unbound applet into an initiate state and then into a start state, respectively.

2. (Original) The method of claim 1, further comprising:
if the unbound applet is completed, issuing predetermined commands to the virtual machine to set the unbound applet into a stop state and into a destroy state, respectively.

3. (Original) The method of claim 1, further comprising:
if determined that the applet is a bound applet and determined by the browser that a markup document connected to the bound applet has a grammatically correct structure, issuing a predetermined command to the virtual machine to set the bound applet into an initiate state.

4. (Original) The method of claim 3, further comprising:
issuing a command to the virtual machine each time the markup document is displayed to set the bound applet into a start state while the markup document is being rendered by the browser;

if the markup document is unloaded by the browser, issuing a command to the virtual machine to set the bound applet into a stop state; and

issuing a command to the virtual machine to set the applet into a destroy state to delete

the bound applet.

5. (Currently Amended) An apparatus executing a markup document applet, comprising:

- a memory which stores an input markup document;
- a virtual machine which executes an applet related to the markup document;
- a browser which receives the markup document from the memory and outputs information on ~~an~~the applet related to the markup document included in the markup document;
- and

- an application manager which receives the applet information from the browser, retrieves the applet from an external data source, controls the retrieved applet to be stored in the memory, receives a request for executing the stored applet, loads the stored applet into the virtual machine, determines whether the loaded applet is a bound applet or an unbound applet, and if the loaded applet is ~~an~~the unbound applet, immediately issues predetermined commands to the virtual machine to first set the loaded unbound applet into an initiate state and then into a start state, respectively.

6. (Original) The apparatus of claim 5, wherein when the virtual machine informs the application manager that the unbound applet is completed, and the application manager issues a command to the virtual machine to set the unbound applet into a stop state and issues another command to the virtual machine to set the unbound applet into a destroy state to unload the unbound applet from the virtual machine.

7. (Original) The apparatus of claim 5, wherein if the applet is a bound applet, the browser informs the application manager that the markup document connected to the bound applet has a grammatically correct structure, and the application manager issues a command to the virtual machine to set the bound applet into an initiate state.

8. (Original) The apparatus of claim 7, wherein the application manager further:

- issues a command to the virtual machine to set the bound applet into a start state while the markup document is being rendered by the browser,
- issues another command to the virtual machine to set the bound applet into stop state, if the markup document is unloaded by the browser,
- repeatedly issues the initiate and stop commands to the virtual machine to start and stop

the bound applet upon redisplaying the markup document by the browser, and
issues a command to the virtual machine to set the bound applet into a destroy state to
unload the bound applet from the virtual machine.

9. (Currently Amended) A computer-readable recording medium storing at least one
program controlling an interactive contents reproduction apparatus to execute a markup applet
according to a process comprising:

receiving a request for executing an applet from the browser;
determining whether the requested applet is a bound applet or an unbound applet;
loading the requested applet into a virtual machine; and

if the requested applet is ~~an~~the unbound applet, immediately issuing predetermined
commands to the virtual machine to first set the requested loaded unbound applet into an initiate
state and then into a start state, respectively.

10. (Currently Amended) An interactive digital versatile disc (DVD) player,
comprising:

a programmed computer processor controlling the player according to a process
comprising:

processing a markup document classifying tagged applets into bound and
unbound applets to display interactive contents,

determining whether an applet execution of the markup document is a bound
applet or an unbound applet according to the classifying, and

if the applet is ~~an~~the unbound applet, launching the unbound applet by
immediately issuing predetermined commands to first set the unbound applet into an initiate
state and then into a start state, respectively.

11. (Original) The player of claim 10, wherein the programmed computer processor
launches the unbound applet without synchronization with the markup document processing.

12 (Original) The player of claim 10, wherein the launched unbound applet
continuously executes independent of the markup document processing.

13. (Original) A method, comprising:
classifying tagged applets of a markup document; and

controlling different execution life cycles of the tagged applets according to the classifying.

14. (Original) The method of claim 13, wherein the classifying comprises classifying the tagged applets into bound and unbound applets, and wherein according to the controlling, execution life cycle of a bound applet depends on the markup document life, and the execution life cycle of an unbound applet is independent of the markup document life.